Confidence Measures for Voice Search Applications

Ye-Yi Wang, Dong Yu, Yu-Cheng Ju, Geoffrey Zweig and Alex Acero

Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA

{yeyiwang,dongyu,yuncj,gzweig,alexac}@microsoft.com

Abstract

Voice search is the technology underlying many spoken dialog applications that enable users to access information using spoken queries. This paper reviews voice search technology, and proposes a new and effective method for computing semantic confidence measures. It explores the use of maximum entropy classifiers as confidence models, and investigates a feature selection algorithm that leads to an effective subset of prominent features for the classifier. The experimental results on a directory assistance application show that the reduced feature set not only makes the model more effective in handling different recognition and search engine combinations, but also results in a very informative confidence measure that is closely correlated with the actual voice search accuracy.

Index Terms: voice search, directory assistance, confidence measure, Tf-Idf vector space model, maximum entropy model.

10.21437/Interspeech.2007-68

1. Introduction

The confidence measure in a dialog system depicts the system's level of uncertainty in its interpretation of a user's utterance. It is an important component of a spoken dialog system – the dialog manager relies on it to determine an appropriate conversation strategy. Unlike confidence measures in automatic speech recognition (ASR) [1], the confidence measure of a dialog system needs to take into account the uncertainties from different components in the inference stages that lead to an interpretation of a user's utterance. In [2] and [3], features from ASR and semantic analysis, either knowledge-based or data-driven, have been used in deriving a confidence measure. In [4], features from ASR and classification components are used to derive confidence measures for a call-routing dialog system. This paper addresses the problem of confidence measures in a new type of spoken dialog system, voice search applications.

Voice search is the technology underlying many spoken dialog applications that provide users with the information they request with a spoken query. For example, directory assistance is one of the most popular voice search applications, in which users issue a spoken query to an automated system which returns phone number and address information for a business or an individual.

The characteristics of voice search pose new challenges to spoken dialog technology. A voice search application differs from ATIS [5] style systems. It does not require detailed semantic analysis to obtain the semantic frame and its slots from an utterance, as in [2] and [3]. It differs from call-routing types of applications in the sense that its inventory of "routing destinations" is enormous, sometimes containing hundreds of thousands entries. The available data will seldom be sufficient to train a statistical model like a Maximum Entropy (ME) classifier or boosting algorithms. For ASR, the vocabulary of a voice search system can be much bigger than a typical domain-specific application -- sometimes reaching millions of lexical entries. Voice search further needs to be robust to high ASR error rates (typically around 30~40%), and linguistic diversity in users' queries – users may not know or would not say the exact name of an entry, e.g., users would typically say "Sears Department Store" or "Sears" rather than the technically correct name, "Sears Roebuck & Company."

The remaining part of this section introduces the voice search technology that addresses these challenges. After that, the issues of confidence measures for voice search are discussed in section 2. Section 3 describes our experiments and results, and section 4 concludes the paper.

1.1. Robust voice search technology

A typical voice search system consists of several components – an automatic speech recognizer converts a user's speech input into a query in text form; a search component looks for the entries in an inventory, e.g., businesses in yellow pages, that match the query; a disambiguation component reduces the size of the result set according to any additional information provided by a user; and a dialog manager that controls the flow of the conversation with a user.

In [6], finite state transducers (FSTs) are used as language models (LMs) for ASR. The FSTs are constructed from the "signatures" of business listing names in a database. Since the output from the transducer is the same as the listing names in the database, the spoken language understanding (SLU) can be a simple database lookup to find the information requested by a user. However, this approach is not robust to linguistic diversity and ASR errors.

In our current work, we propose an architecture in which we perform ASR using an n-gram language model trained with database listing names and smoothed with a large vocabulary back-off LM [7], and use the vector space model (VSM) [8] for SLU. The VSM has been widely used in information retrieval. It represents ASR results and listing names as Tf-Idf weighted vectors and finds the relevant listing (document) vector with the highest cosine similarity to an ASR (query) vector. The n-gram LM makes voice search robust to linguistic diversity, and the "fuzzy" matching capability of VSM makes it robust to ASR errors and linguistic diversity. Our internal studies indicate that it significantly outperforms the FST-based approach.

Listings in a database are often associated with category information, e.g., "restaurant" or "healthcare," in a business database, or "electronics" or "DVDs" in a product database. To further improve search robustness, cosine similarity based on listing names is interpolated (smoothed) with a category similarity: $\text{sim}(Q,L) = \alpha \cos(Q,L) + (1-\alpha)\cos(Q,C(L))$. Here $C(L)$ is the category of L in a database. It is represented as a vector of the document that contains all the listing names of that category. In doing so, a query like "Overlake hospital" is more likely to match the listing "Overlake Medical Center" than the listing "Overlake café," because the former is of the

"healthcare" category and many listings in that category have "hospital" in their names.

2. Binary Classification Models for Confidence Measure

2.1. Confidence measure as a classification task

Given a user's spoken query Q and the database listing L found by a voice search system as its answer, a confidence model has to decide how likely L is the correct answer, based on some supporting statistics (features) collected from the process leading to the finding of L. A confidence score of continuous value is often used by a dialog manager to adopt different response strategies at different confidence levels according to a designer's specification. A binary statistical classifier assigns a probability to the CORRECT and INCORRECT decisions (classes). The probability of the CORRECT class can be used as the confidence score.

2.2. Maximum entropy classification

A Maximum Entropy (ME) classifier [9] is a discriminative model that generally yields better classification results than a generative model. It builds the conditional probability distribution $\mathrm{P}(C \mid Q,L)$ from a set of features $\mathcal{F}$, where C is a random variable representing the classification destinations. In the case of confidence modeling, the range of the variable is {CORRECT, INCORRECT}. Q and L are random variables representing the spoken query and the database listing, respectively. A feature in $\mathcal{F}$ is a function of C, Q and L. The classifier picks a distribution $\mathrm{P}(C \mid Q,L)$ to maximize the conditional entropy $H(C \mid Q,L)$ from a family of distributions, with the constraint that the expected count of a feature predicted by the conditional distribution equals to the empirical count of the feature observed in the training data:

$$\sum_{C,Q,L} \tilde{\mathrm{P}}(Q,L)\cdot \mathrm{P}(C \mid Q,L)\cdot f_i(C,Q,L) = \sum_{C,Q,L} \tilde{\mathrm{P}}(C,Q,L)\cdot f_i(C,Q,L), \quad \forall f_i \in \mathcal{F} \tag{1}$$

where $\tilde{\mathrm{P}}$ stands for empirical distributions over a training set.

It has been proven that the maximum entropy distributions that satisfy Eq. 1 have the following exponential (log-linear) form [9]:

$$\mathrm{P}(C \mid Q,L) = \frac{1}{Z_\lambda(Q,L)} \exp\left(\sum_{f_i \in \mathcal{F}} \lambda_i f_i(C,Q,L)\right), \tag{2}$$

here $Z_\lambda(Q,L) = \sum_C \exp(\sum_{f_i \in \mathcal{F}} \lambda_i f_i(C,Q,L))$ is a normalization constant, and λ_i's are the parameters of the model. They are also known as the weights of feature f_i, which can be optimized using training data. For model training, we applied the stochastic gradient descent algorithm [10].

2.3. Features

Features are extracted from the ASR, SLU (search) and dialog manger components for the ME classification model. Below is a summary of the candidate features that have been considered:

2.3.1. ASR features

We included ASR confidence and ASR semantic confidence in the feature set. The former is the confidence measure from the ASR on the entire utterance, and the latter measure the confidence of the semantic content (associated with the semantic tags in W3C standardized SRGS grammar [11].) To make the confidence model applicable to voice search systems with different ASR engines, we chose to not include those features that are not broadly available in commercial recognizers, such as lattice density used in the referenced papers. To a certain degree, those features have already been encapsulated in the ASR confidence measures.

2.3.2. Search features

Given a query Q and a hypothesized listing L, i.e., the listing with the highest category smoothed vector similarity with Q, the search related features include the Tf-Idf weighted vector similarity between Q and L, with or without category smoothing (henceforth Tf-Idf score (Category) and Tf-Idf score (No Category)); the gap between the unsmoothed similarity score of L and the highest unsmoothed vector similarity (Tf-Idf gap), which might be greater than 0 if the highest unsmoothed score is registered with a listing other than L; the ratio between the maximum Idf value among the words existing in both Q and L and the maximum Idf value among all the words in L (Covered/Uncovered Idf ratio); and the number of matching characters in Q and L, normalized by the query and listing lengths: $M^2/|Q||L|$. (Normalized character matches.) Here M can be obtained with dynamic programming.

2.3.3. Dialog manager features

The dialog manager features include dialog turn, previous turn occurrence, and city match. The first is an integer that represents the dialog turn at which a spoken query was issued. The second is a binary variable that is activated if the listing returned by the search component has been hypothesized and presented to the user in a previous dialog turn and rejected by the user. The third is an application specific feature — it is activated in a directory assistance system if the city of the hypothesized business listing matches the city specified by the user at the beginning of a dialog.

2.3.4. Combined features

Combined features attempt to model the dependency among features across different components in voice search. They include the ASR confidence on the individual word that also exists in L and has the highest Idf value, i.e. the ASR confidence of the word that contributes the most to the search result (Confidence of max. Idf word). Another combined feature is the joint of ASR sentence confidence and the smoothed Tf-Idf score, whose value set is the Cartesian product of the value sets of the two features.

Many features described above have continuous values (e.g., ASR confidence, Tf-Idf score, etc). While the ME classifier, a log-linear model, can take continuous features, it assumes linear relation between feature values and the class boundary, which seldom holds in this case. Figure 1 plots the percentage of CORRECT samples as a function of feature values for 4 different continuous features and shows nonlinearity for all the features. Therefore, instead of using continuous features, we quantize the features into 20 evenly distributed discrete buckets, and each bucket is represented by

a binary random variable that has value 1 if a continuous feature falls into the bucket.

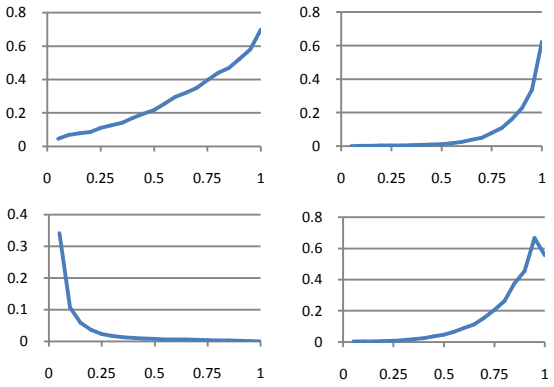

Figure 1. *End-to-end accuracy vs. feature values in a directory assistance system. Top left: ASR confidence feature. Top right: Tf-Idf score feature. Bottom left: Tf-Idf gap feature. Bottom right: normalized character match feature.*

3. Experiments

3.1. Data

A pilot directory assistance system has been deployed to a user group for data collection. The system logs the relevant information necessary to extract the features for each pair of utterance Q and the hypothesized listing L. The training and testing of the ME classifier requires an assignment of CORRECT or INCORRECT class label to each pair. The labels were automatically assigned according to the users' response to the following confirmation prompt (e.g., "You are looking for the number for Macy's, is it correct?"). The users' responses to the confirmation prompts were manually corrected for ASR errors. We partitioned the data into training (~700 pairs), development (~300 pairs) and test (~300 pairs) sets. The database that the system searches against contains ~18 million business entries.

3.2. Classification results

Table 1 shows the error rate of the ME classifier on the binary classification task. The end-to-end error rate on the directory assistance task is around 36%. So the baseline (chance) error rate on classification is 36% if the system always makes a guess of CORRECT class. In contrast, the ME classifier that uses all the features cut the errors by 50%.

Table 1. *Binary classification error rates on the development and test set.*

Dev set	Chance	36.91%
	ME classifier using all features	18.46%
Test set	Chance	36.21%
	ME classifier using all features	17.94%

3.3. Experiments on feature selection

In the second experiment, we investigated the importance of different features on the classification accuracy, and selected a subset of features for our final confidence model. With fewer features, the model has fewer parameters and is less subject to over-fitting. In addition, the model is more practical because it poses fewer requirements on the voice search components to report different statistics. For that purpose, we studied the impact on the classification error rate by removing individual features from the feature set. Table 3 shows the development set error rates after the removal of individual features from the feature set, as well as the significance of the change inflicted by the removal of the feature (the null hypothesis probability in a sign test). According to it, the removal of Tf-Idf score without category smoothing is least significant (with null hypothesis probability 1.0). So this feature is removed to form an updated baseline and the experiment is repeated, with the new results listed in Table 3.

From Table 3, we selected a subset of features whose removal inflicts a big performance change, in the sense that the null hypothesis probability is smaller than 0.5. The subset has 5 features, namely the ASR semantic confidence score, the ASR confidence on the word with the highest Idf value in the match against a listing, the category smoothed Tf-Idf score, the normalized character matches, and the Tf-Idf gap. Table 4 compares the error rate with and without feature selection on the test set, which shows no significant difference. Because there are no application dependent features in the subset, the confidence model can be applied to different voice search applications.

Table 2. *The development set error rate and significance of change* (*probability of null hypothesis*) *after the removal of individual features.*

Features	**Err. Rate**	**P**(null)
All (baseline)	18.46%	
All-City match	20.47%	0.11
All-Dialog turn	20.47%	0.13
All-Normalized Character Matches	19.80%	0.26
All-ASR confidence	21.48%	0.02
All-ASR semantic confidence	20.81%	0.06
All-Joint ASR confidence/Tf-Idf score	20.81%	0.07
All-Covered/uncovered Idf ratio	19.80%	0.21
All-Confidence of max Idf word	19.13%	0.40
All-Prev turn occurrence	20.47%	0.11
All-Tf-Idf gap	19.80%	0.21
All-Tf-Idf score (Category)	20.81%	0.13
All-Tf-Idf score (No category)	18.46%	1.00

Table 3. *The development set error rate and significance of change* (*probability of null hypothesis*) *after the removal of individual features from the updated baseline feature set.*

Features	**Err. Rate**	**P**(null)
Base = All - Tf-Idf score (No Category)	18.46%	
Base-City match	18.46%	1.00
Base-Dialog turn	17.79%	0.86
Base-Normalized character matches	19.13%	0.41
Base-ASR confidence	18.79%	0.50
Base-ASR semantic confidence	20.81%	0.03
Base-Joint ASR confidence/Tf-Idf score	18.46%	0.64
Base-Covered/uncovered Idf ratio	18.46%	0.75
Base-Confidence on max Idf word	19.46%	0.25
Base-Prev turn occurrence	18.46%	1.00
Base-Tf-Idf gap	19.80%	0.11
Base-Tf-Idf score (Category)	19.13%	0.43

Table 4. *Classification error rate with the selected features.*

Test set	ME classifier using all features	17.94%
	ME classifier using selected features	17.60%

3.4. Confidence measure experimental results

Binary classification is not the actual objective of confidence measures. A dialog manager often sets the thresholds for high, medium and low confidences, obtains a numeric score from

the confidence model and bases its decision upon what confidence interval the score falls into. We used the conditional probability P(CORRECT | *Q, L*) returned by the classifier as the confidence score.

Figure 2 depicts the ROC curves for the acceptance rate of the positive data (i.e., voice search returns correct results) and the rejection rate of the negative data in relation to the acceptance threshold – a test case is accepted as CORRECT only when its confidence score is higher than the threshold. For example, at the intersection of the two curves (threshold ≈ 0.6), the confidence measure accepts 80% of the correct search results and rejects 80% of incorrect search results. Based on this figure, dialog manager designers can select different thresholds according to their application scenarios.

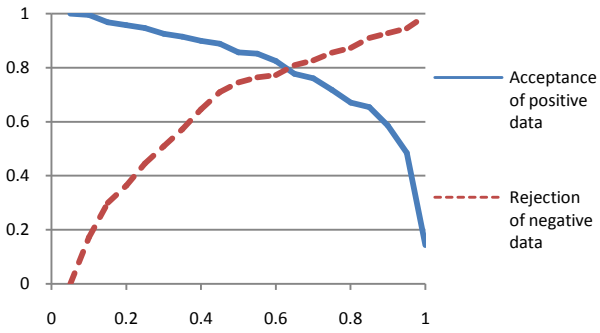

Figure 2. *ROC curves for the acceptance of the* CORRECT *test data and the rejection of the* INCORRECT *test data. The X-axis represents the acceptance threshold, and the Y-axis represents the recall of correct acceptances/rejections.*

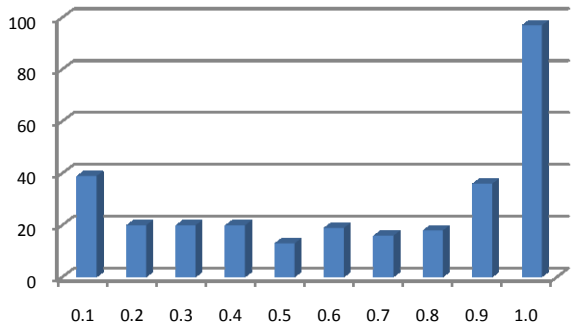

Figure 3. *Distribution of test data in different confidence intervals.*

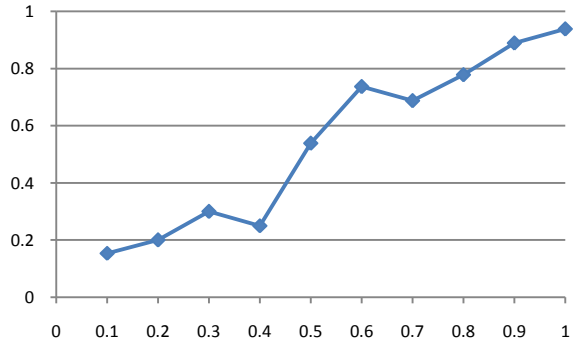

Figure 4. *Correlation between the confidence measure and the actual voice search accuracy. X-axis represents the confidence intervals; Y-axis represents the test set accuracy.*

A confidence measure is not very informative if it reports a borderline confidence (around 0.5) most of the time. Figure 3 shows that the proposed confidence model assigns either very high (>0.8) or very low (< 0.2) confidence scores to most of the test data.

Finally, a confidence measure should reflect the true end-to-end accuracy – the higher the score it assigns to a test case, the more likely the test case get the correct search result. Figure 4 illustrates the accuracy of test data at different confidence intervals, which shows a strong correlation between the confidence measure and the actual voice search accuracy.

4. Conclusions

Voice search applications bear their own characteristics that require different technological solutions. A viable solution combines robust ASR and a robust search algorithm. The maximum entropy classifier has been successfully applied to derive confidence measure in the framework of voice search. We have shown that feature selection can reduce the number of features without sacrificing accuracy, and hence made the confidence measure more effective in handling different recognition and search engines. The experimental results show that the confidence measure is very informative and has a good correlation with the actual end-to-end voice search accuracy.

5. References

1. Wessel, F., et al., *Confidence measures for large vocabulary continuous speech recognition.* IEEE Transaction on Speech and Audio Processing, 2001. **9**(3).
2. Sarikaya, R., et al., *Semantic confidence measurement for spoken dialog systems.* IEEE Transactions on Speech and Audio Processing, 2005. **13**(4): p. 534- 545.
3. San-Segundo, R., et al., *Confidence Measures for Spoken Dialog Systems*. In the proceedings of the *IEEE International Conference on Acoustics, Speech, and Signal Processing* 2001.
4. Walker, M., et al., *Learning to Predict Problematic Situations in a Spoken Dialogue System: Experiments with How May I Help You?* In the proceedings of the *Meeting of the North American Chapter of the Association for Computational Lingusitics*. 2000.
5. Price, P. *Evaluation of Spoken Language System: the ATIS domain*. In the proceedings of *DARPA Speech and Natural Language Workshop*. 1990. Hidden Valley, PA.
6. Jan, E.E., et al., *Automatic Construction of Unique Signatures and Confusable Sets for Natural Language Directory Assistance Applications*, in *Eurospeech*. 2003: Geneva, Switzerland.
7. Yu, D., et al. *N-Gram Based Filler Model for Robust Grammar Authoring*. In the proceedings of *International Conference on Acoustics, Speech, and Signal Processing*. 2006. Toulouse, France.
8. Salton, G., *Automatic Information Organization and Retrieval*. 1968, New York: McGraw-Hill.
9. Berger, A.L., S.A. Della Pietra, and V.J. Della Pietra, *A Maximum Entropy Approach to Natural Language Processing.* Computational Linguistics, 1996. **22**(1): p. 39-72.
10. Kushner, H.J. and G.G. Yin, *Stochastic Approximation Algorithms and Applications*. 1997: Springer-Verlag.
11. Hunt, A. and S. McGlashan, *Speech Recognition Grammar Specification Version 1.0*. http://www.w3.org/TR/speech-grammar. 2002.